|  |  |
| --- | --- |
| **Politechnika Śląska** | **Gliwice** |
| **Wydział Automatyki Elektroniki i Informatyki**  **Kierunek: Automatyka i Robotyka sem. 5** | **Rok akademicki 2019/2020** |
|  |  |

**Systemy Operacyjne**

**Laboratorium**

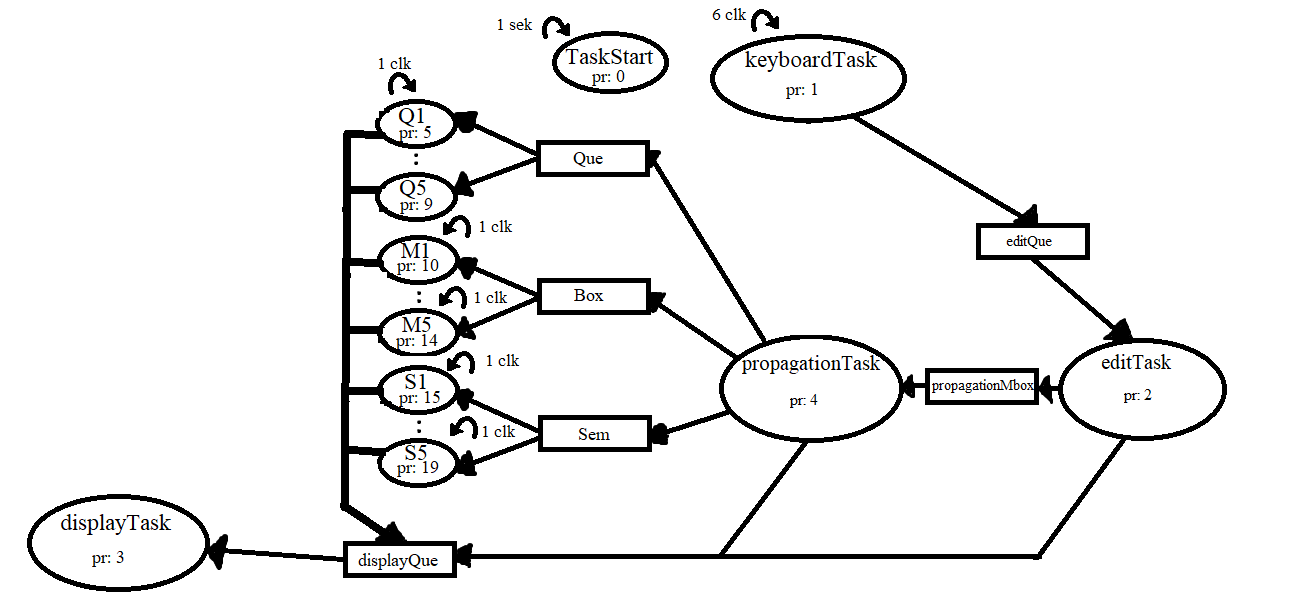
**RTOS**

**Wykonali:**

**Dawid Mudry**

**Karol Marciniak**

**Grupa 1**



|  |  |  |  |
| --- | --- | --- | --- |
| BufforSize | Priorytet | Nazwa | Funkcje |
| 1024B | 0 | StartTask | Tworzy wszystkie taski, updatuje (aktualizuje) ekran |
| 1024B | 1 | keyboardTask | Sczytywanie klawiszy |
| 1024B | 2 | editTask | Interpretacja klawiszy i tworzenie buffora |
| 1024B | 3 | displayTask | Wyświetlanie informacji |
| 1024B | 4 | propagationTask | Rozsyła informacje, wykrywanie błędów |
| 1024B | 5-9 | QueTask | Zadanie obciążające używające kolejek |
| 1024B | 10-14 | BoxTask | Zadanie obciążające używające Mbox |
| 1024B | 15-19 | SemTask | Zadanie obciążające używające semafora |
| | | | | | | | |
| 1024B | 21 | IdleTask | Idletask |

*Wykres 1.*

Procesor: IntelCore i5-7200U, 2,5 GHz

*Wykres 2.*   
Procesor: IntelCore i5-7200U, 2,5 GHz

Context switch oraz zużycie procesora zostało przeskalowane dzieląc odpowiednio przez 100 oraz 3, aby dopasować wartości do wykresu.

Interpretacja wykresów:

*Wykres 1*:

Wraz ze wzrostem obciążenia maleje eksponencjalnie contex switch, dla dane odcinka wykresu zużycie procesora pozostaje cały czas na poziomie 100 procent, ilość zadań pracujących również maleje eksponencjalnie od pewnej wartości która została pokazana na wykresie logarytmicznym

*Wykres 2*:

Wraz ze wzrostem obciążenia rośnie zużycie procesora, aby dla obciążenia około 350000 osiągnąć 100 procent, od tamtego momentu zaczyna maleć ilość zadań które pracują w systemie.

Context switch wraz ze wzrostem obciążenia maleje, Dla obciążenia 200000000 pracuje już tylko jedno zadanie system zostaje zapchany.

Napotkane problemy:

Błędy podczas uruchamiania programu:

Napotkano niedozwoloną instrukcje, błąd był związany z wyjściem indeksu poza tablice podczas wywoływania pętli.

Problem z debugowaniem programu wyświetlaliśmy wartości zmiennych za pomocą funkcji PC\_strDisplay() .

Problem związany z taskiem klawiatury, który zbyt szybko ponownie został wywoływany. Fizyczne możliwości sprzętu były zbyt niskie, aby mogły działać poprawnie w związku z tym zwiększyliśmy delay (opóźnienie) taskaKlawiatury zmniejszając użycie procesora przy czym wydajność klawiatury nie zmieniła się.

Błąd związany z większą ilością tasków niż było zadeklarowane w pliku OS\_CFG.H. Rozwiązaliśmy problem zmieniając wartości w kilku miejscach.

Utrata informacji podczas zmieniania obciążenia, była spowodowana wrzucaniem do kolejki, mailboxa tego samego wskaźnika.

Problem z implementacją pamięci dynamicznej w programie, w wywołaniu funckji OSMemGet() w której jednym z argumentów był wskaźnik , niestety przez nieuwagę podawany był wskaźnik wskaźnika co powodowało błąd w wywołaniu i zamkniecie się programu, błąd spowodował poświecenie sporo czasu na debugowanie programu.